

練習寫程式時的要求

NTOUCSE

丁培毅

1

怎麼說這個?! 一板一眼的

- 三十幾年來, 這是一個成功率不高的課程!!

考試掛帥的競爭式的學習環境

所有人都是你的敵人

- 台灣資訊產業的未來在哪裡??

- 每一位路過這裡的同學都是台灣未來的希望與機會, 看你能不能藉由這裡的資料突破自己直覺上以為的限制

合作式的學習環境

你和老師合作、你和同學、學長合作



2

練習寫程式時的基本要求

- 要求你要做到:

「看過/分析過/想過 問題和範例程式,

遮起來, 在電腦上打出來

測試執行, 觀察結果, 分析整理與修改」

- ~~不是 拷貝貼上 ⇒ 測試 沒有錯 ⇒ 完成~~

~~只學到了怎樣使用 鍵盤+滑鼠 操作!!~~

~~錯了 ⇒ { 拷貝別的
隨便改改 } ⇒ 測試, 說不定錯誤會不見~~

3

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int i,n,j;
    const char* const day[12]={"first","second","third","fourth",
        "fifth","sixth","seventh","eighth",
        "ninth","tenth","eleventh","twelfth"};
    const char* const line[11]={"2 Turtle Doves","3 French Hens",
        "4 Calling Birds","5 Golden Rings","6 Geese a Laying",
        "7 Swans a Swimming","8 Maids a Milking","9 Ladies Dancing",
        "10 Lords a Leaping","11 Pipers Piping","12 Drummers Drumming"};
    scanf("%d",&n);
    for (i=0;i<n;i++) {
        if (i>0) printf("\n");
        printf("On the %s day of Christmas\n", day[i]);
        printf("my true love sent to me:\n");
        for (j=12-i; j<12; j++) printf("%s\n", line[11-j]);
        if(i==0) printf("A"); else printf("and a");
        printf(" Partridge in a Pear Tree\n");
    }
    return 0;
}
```

4

• 什麼叫做 背

白居易的「琵琶行」

88句616字

潯陽江頭	江州司馬	浼衫青馬	司州江	多最誰	下泣中	座
主人下馬	滿座重	泣掩皆	聞重座	滿	聲前向	似不淒淒
醉不成	卻坐促	急轉弦	弦促坐	卻	立久良	言此我感
忽聞水	為君翻	行琵琶	作翻君	為	曲一彈	坐更辭莫
尋聲暗	如聽仙	明暫耳	樂仙聽	如	語琵琶	君聞夜今
移船相	猶抱琵	面遮半	琵琶抱	猶	來出始	喚萬呼千
千呼萬	添酒回	宴開重	燈回酒	添	見相邀	近相船移
今夜聞	琵琶聲	遲語欲	停聲琵	琶	誰者彈	問暗聲尋
莫辭更	主人忘	發不客	歸忘人	主	聲琵琶	上水聞忽
感我此	別時茫	月浸江	茫茫時	別	別將慘	歡成不醉
淒淒不	舉酒欲	弦管無	飲欲酒	舉	船在客	馬下人主
座中泣	楓葉荻	瑟瑟秋	花荻葉	楓	客送夜	頭江陽潯

9

考試型的學習

• 套公式就好

計算 $4 + 5 + 6 + 7 + \dots + 50$

為什麼？

$$4 + 5 + 6 + 7 + \dots + 50$$

$$\frac{50 + 49 + 48 + 47 + \dots + 4}{54 \quad 54 \quad \quad \quad 54}$$

47 組

怎麼會有怪物這麼想呢？

$$4 + 5 + 6 = 4 + 6 + 5$$

$$\frac{(4 + 50) 47}{2}$$

背

為什麼要知道為什麼呢？ 對付各種變化

常常練習，就算沒有人告訴你步驟，自己整理出來也不是太困難

10

寫程式是在說故事、在講話、

用固定格式的指令操控機器做許多事情

重點是你想講什麼、你要叫電腦做什麼？

如果你是公司老闆，永遠搞不清楚下屬名字和

部門的分工，沒有名單就無法指揮，這樣會很好笑，沒有什麼說服力吧!! 很難專心在公司的營運目標吧!!

練習寫程式的時候，除了格式要符合規定，重點是思考程式的內容，一旦你照著資料抄，甚至連格式都抄，的確可以省下很多時間，但是常常也分不清楚什麼是格式、什麼是內容了，通常你只會記得去哪裡抄的...

這樣很不好，你才剛進來

不小心就開始跟資訊業說 byebye

11

- 程式有固定的格式：每一段程式功能可以不同，但是需要遵循一定的規則 – 有些部份是規定要有的

```
#include <stdio.h>
int main() {
    printf("Hello World!\n");
    return 0;
}
```

- 像是英文裡面的語法：主詞 + 動詞 + 受詞 + 副詞
- 後續會逐步介紹 C 程式的語法
- 不要過度擔心：人和機器溝通的語法比人與人溝通的語法簡單得多，規則少、沒有背不完的單字、也沒有因為習慣而產生許許多多的例外

12

- 你需要練習到不需要查書、不需要查資料, 就可以正確使用基本的語法, 你的腦筋才能夠專注在該放什麼命令, 該以什麼順序來下達命令上面
- 這就是先想再打練習的主要目標
- **Q:** 重頭打一遍好浪費時間, 都在找按鍵在哪裡
這時候太想要節省時間, 又害你走不進資訊系的大門
 - 長久 ... 練習英文打字吧! (不是高職生的專利)
 - 暫時 ... 一列一列拷貝 ... (認真的)
你在心裡問自己: 接下來要寫什麼?
是 int, 是 (, 是 }, 是 return, 還是其它敘述?
想清楚了, 確定是什麼了, 驗證一下, 才拷貝一列

13

- **Q:** 看過範例程式了, 覺得再怎麼看也沒有幫助了

- 可是遮起來以後一點點都寫不出來
- 範例不好: 新識不出它的架構

```
void bucketSort(int ndata, int data[]) {
    int buckets[8][MAX], nBucket[8];
    int i, j, k, index, mult=1, iBucket;
    int len = maxNumDigits(ndata, data);
    for (i=0; i<len; i++) {
        for (j=0; j<8; j++) nBucket[j] = 0;
```

```
#include <stdio.h>
#define BING(x,y) ((x)<<y)
#define BANG(x) (1<<x)
#define BOOM 1
int main () {
    int x,y,z,w;
    int V[3] = {BING(x=227380393,BANG(BOOM)+BOOM), x+(w=BOOM+BANG(BANG(BOOM))),
                BING(47*(y=17453197),BOOM)), x+y+BING(w*w*17*185527,BANG(BOOM))};
    char *p=(char *)V;
    while(*(p-BOOM)!=BOOM+BING(BOOM,w)) putchar(*p++);
    return 0;
} /* Mind the comma operator! */
```

```
q:\>g++ prog.cpp -o prog.exe
q:\>prog
Hello World!
```

14

- 有看沒懂: 沒有還原出程式背後運作的模型

- 想像一下你現在要幫一個由中港大到淡江的指引

- 有沒有出現北寧路公車站的樣子
- 有沒有出現103/104公車上擁擠的樣子
- 有沒有出現基隆車站的樣子
- 有沒有出現電聯車上的樣子
- 有沒有出現北車大迷宮的樣子
- 有沒有出現捷運淡水車站的樣子
- 有沒有出現捷運淡水車站的樣子
- 有沒有出現等候公車站的樣子

要教會別人之前, 心中通常有很清楚的經驗圖像

15

- 有看沒懂: 沒有還原出程式背後運作的模型

需要了解抽象

- 由小到大排好 n 個

- e.g. 100, 003, 6

- 如下圖準備八列空

	0	1	2		n-1	
0	100	120				
1						
2						
3	003	013		...		
4						
5	325					
6						
7	667	027				

100|120|003|013|325|667|027

```
void bucketSort(int ndata, int data[]) {
    int buckets[8][MAX], nBucket[8];
    int i, j, k, index, mult=1, iBucket;
    int len = maxNumDigits(ndata, data);
    for (i=0; i<len; i++) {
```

100|003|013|120|325|027|667

003|013|027|100|120|325|667

16

協助編碼的工具

- 撰寫程式時還是需要記住好多東西, 都沒有工具可以幫忙嗎?
 - 編譯器: 檢查你寫的程式是否符合語法
 - 自動程式碼完成工具: Intellisense (MS), Eclipse, Code completion plugin of Code::Blocks 主要是提示函式名稱, 函式參數, 語法一寫錯就立刻提示

```
int i, num = -1, K = -1;
char digit[MAXNUMBER];
scanf("%d%d", &num, &K);
Error: 識別項 "num" 未定義
```

```
for (i=0; i<num; i++) digit[i] = i;
do
    if (checkOutOfOrderCounts)
Error: 識別項 "checkOutOfOrderCounts" 未定義
```

- 線上語法說明
- google

```
do
    if (checkOutOfOrderCountsK(num, digit, K))
        int checkOutOfOrderCountsK(int size, char *data, int K)
while (r
return 0;
```

17

協助編碼的工具 (cont'd)

- 太開心了, 有了這些工具, 語法不必記下來了吧? 有更重要的東西需要我的腦袋!!
 - 編譯器是最基本的工具, 一定要使用
 - 錯誤訊息一定要仔細看懂
 - 不要太依賴自動程式碼完成工具
 - 很多開發環境中不提供
 - 比較複雜的程式架構中建議會錯
 - 對語法越熟悉, 撰寫程式時才可以專心構思程式該有的功能, 不會一直被不確定的語法干擾, 隨時可以評估一個想法是否寫得出來

18

練習寫程式的三大殺手

- 及格就好: 不是 60 分就及格了嗎?!
 - 一個程式有一點點(1%)邏輯錯誤時, 輸出的答案就是錯誤的, 對於使用這個程式的人是沒有用的
- 二分法: 以為程式不是對的就是錯的
 - 以為程式發生錯誤時, 所有的過程都是沒有用的!!
 - 換一個程式, 換一個方法... 換一個系 😞
 - 對使用程式的人沒有用, 對撰寫這個程式的人可是有非常大的用途的
- 考試型的學習, 效率掛帥:
 - 以為程式發生錯誤時, 正確答案在課本上, 換成正確的就好, 省時省力

debug: 分析目前的程式和結果, 一點一點修改程式來得到想要的結果

19

實習課程的特點

- 大部分實習課程只說明語法或是只給程式題目/答案
- 直接讓你看問題的解決範例程式或是虛擬程式碼
- 可是學習設計程式時最大的障礙在於
 - 為什麼會想到運用某一種語法來解決問題
 - 怎麼簡化與分析問題, 一步一步得到解決問題的方法
- 只有題目, 同學很難進入狀況; 直接給虛擬程式碼或是程式碼, 練習的時候其實已經沒有思考變化的空間了
- 所以在這個課程以及實習裡, 我盡量避免直接讓你看到最後的程式碼, 而是盡量一步驟一步驟說明思考的重點、需求、方向、以及轉折點
- 不直接讓你看到程式碼, 目的不在吊你胃口, 也不是不考慮你其他課程的負擔, 而是希望問題在你的腦中成型, 由多個角度去辨認這個問題, 抽象化這個問題, 從而思考解決方案

20